BERT

0. Abstract

- BERT는 0기존의 언어 표현 모델과 달리, unlabeled text로부터 모든 층에서 좌우 문맥을 동시에 고려하는 **deep** bidirectional representations이 가능함
- 즉, unlabeled data로부터 pre-training을 진행한 후, 이를 특정 downstream task(with labeled data)에 fine-tuning(transfer learning)을 하는 모델
- BERT는 별도의 task-specific 변형 없이, 하나의 layer만 추가해도 대다수의 NLP tasks에서 좋은 성능을 보임

1. Introduction

사전 훈련된 언어 표현을 downstream task에 적용하는 기존 접근법은 크게 두 가지

1. feature-based 접근법
   -> ELMo는 사전 훈련된 표현을 추가적인 특징으로 활용하는 task-specific 구조를 사용
2. fine-tuning 접근법
   -> GPT는 최소한의 task 별 파라미터만 도입하고, 모든 사전 훈련된 파라미터를 downstream task에서 직접 미세 조정

- 두 접근법 모두 사전 훈련 과정에서 같은 목적 함수를 공유, 일반적인 언어 표현 학습을 위해 단방향 언어 모델 사용
- 위의 두 기존 기법이 사전 훈련된 표현의 성능을 제한함
  -> 특히 언어 모델이 단방향성을 갖는다는 것(=이전 토큰만을 참조)은 질의응답과 같은 단어 수준 task에서는 양방향 문맥이 필수적이므로 심각한 성능 저하 초래할 가능성
- BERT를 이용하여 기존의 fine-tuning 접근법 개선
- Masked Language Model을 활용한 사전 훈련 목표를 도입함으로써 기존 단방향성 문제 해결 : 일부 토큰을 무작위로 마스킹한 후 문맥 정보를 이용하여 마스킹된 원래 어휘 예측
  -> 좌우 문맥을 모두 통합하여 깊은 양방향 트랜스포머를 학습할 수 있음
- 양방향 사전 훈련의 중요성 입증
- 사전 훈련된 표현이 복잡한 태스크별 아키텍처의 필요성을 줄일 수 있음

2. Related Work

2.1 Unsupervised Feature-based Approaches

- 단어 혹은 문장의 representation 학습은 non-neural method와 neural method로 나뉨
- pre-trained된 word embeddngs는 현대 NLP의 필수 요소로 자리 잡음
- word embeddings를 통한 접근 방식은 sentence embedding 혹은 paragraph embedding으로 이어짐
- BERT 이전까지의 sentence representations 학습은
  1. 다음 문장의 후보들을 순위 매기는 방법
  2. 이전 문장이 주어졌을 때, 다음 문장의 left-to-right generation 방법
  3. denoising auto-encoder에서 파생된 방법

- ELMO와 이전 모델은 left-to-right와 right-to-left 언어 모델을 통해 context-sensitive feature 뽑아내는 방식
  -> 생성된 토큰 별 contextual representaition은 left-to-right와 right-to-left representation의 단순 concat
  -> 이는 feature-based 접근법이며, deep bidirectional 하지 않음

2.2 Unsupervised Fine-tuning Approaches

- 초기 연구는 unlabeled text에서 word embedding만을 pre-training 하는 방식
- 최근 연구는 sentence/document encoders를 활용하여 contextual token representations를 학습한 후, supervised task에 맞춰 fine-tuning 하는 방식
  -> scratch로 학습하는 데 필요한 매개변수의 수를 줄일 수 있음

2.3 Transfer Learning from Supervised Data

- supervised learning 데이터셋에서 transfer learning을 수행하는 연구도 효과적인 방법 : CV 연구에서도 대규모 사전 훈련된 모델을 활용한 전이 학습의 중요성 입증

3. BERT

- BERT 프레임 워크는 pre-training과 fine-tuning 두 단계로 구성
- pre-training 단계에서 다양한 사전 학습 태스크를 통해 레이블이 없는 데이터를 학습
- fine-tuning 단계에서는 사전 학습된 모델을 초기화한 후, downstream task의 레이블이 있는 데이터를 사용하여 모든 파라미터를 미세 조정
- 모든 다운스트림 태스크는 동일한 사전 학습된 파라미터로 초기화되지만, 각각 별도의 미세 조정된 모델을 가짐
- Unified architecture를 사용함
  -> 사전 학습된 모델과 최종 다운스트림 모델 간의 차이가 거의 없음

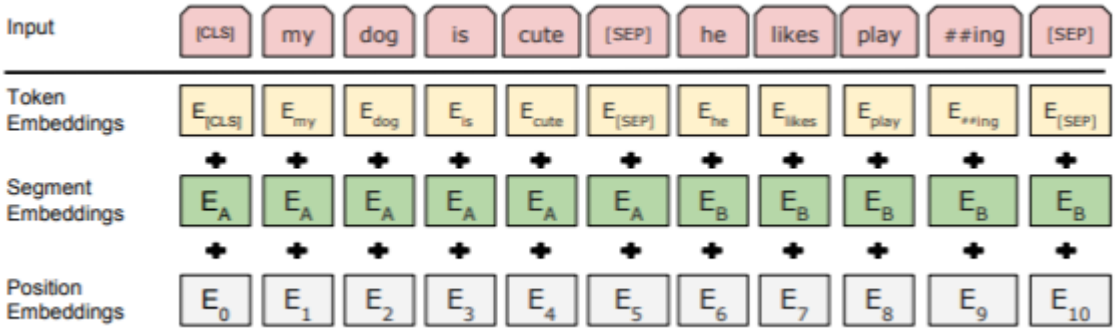
Model Architecture

- 다층 bidirectional transformer encoder로 구성
- 하이퍼 파라미터
  L : 트랜스포머 블록 수(레이어 개수)
  H : 은닉 크기
  A : self attention 헤드 수
- BERT 모델은 두 가지 크기로 제공
  그 중 L=12, H=768, A=12, 총 파리미터 1.1억 개인 $BERT_{BASE}$ 모델은 GPT와 동일한 크기로 설정되었지만, GPT는 좌->우로만 self attention을 수행하는 반면에 BERT는 완전한 양방향 self attention을 수행

Input/Output Representations

- BERT는 다양한 다운스트림 태스크를 처리하기 위해 input representation을 단일 문장과 문장쌍(Q&A)을 하나의 토큰 시퀀스로 애매하지 않게 표현
- 'sentence'는 언어학적인 문장의 의미 뿐만 아니라 인접한 텍스트들의 임의의 범위라는 뜻도 포함
- 모든 입력 시퀀스의 첫 번째 토큰은 [CLS] 토큰
  -> 이 토큰에 해당하는 최종 은닉 상태는 분류 문제를 위해 sequence representation들을 종합함
- 입력 시퀀스는 문장의 한 쌍으로 구성되며, 문장 쌍의 각 문장들은 [SEP] 토큰으로 분리
  -> 각 문장이 A 문장인지, B 문장인지 구분하기 위한 Segment Embedding을 진행

- Token Embeddings는 WordPiece embedding을 사용하고, Position Embeddings는 Transformer에서 사용한 방식과 동일
- Input representation은 이러한 대응되는 토큰(segment + token + position)을 전부 합치면 됨



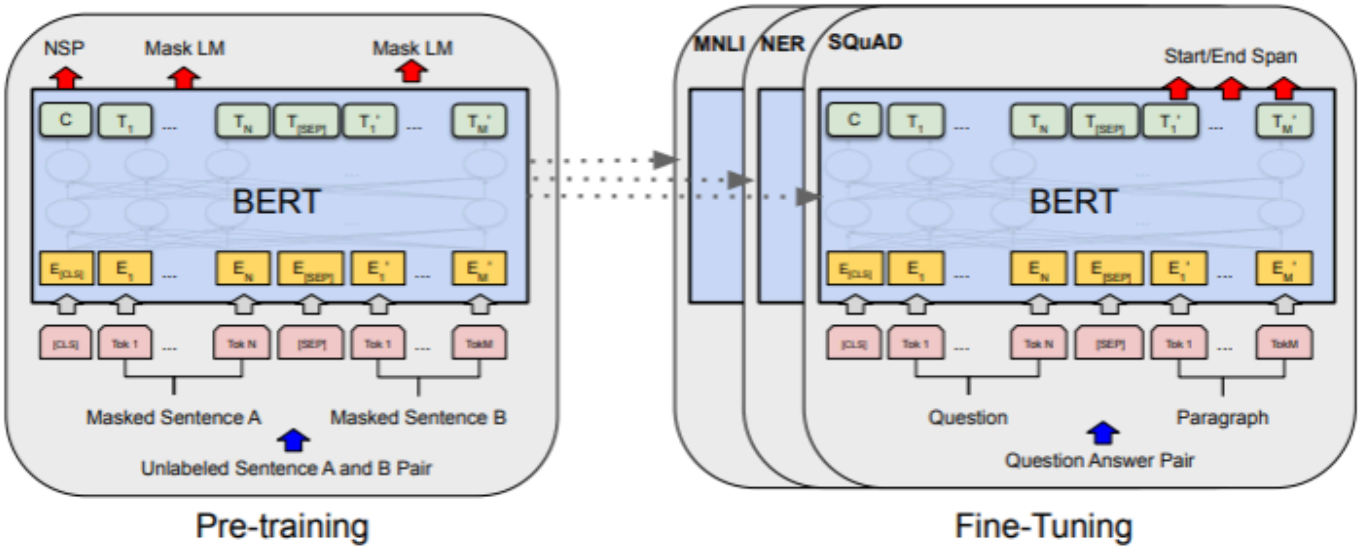
3.1 Pre-training BERT

두 가지 비지도 학습 과제를 사용하여 BERT를 사전 훈련

Task #1: Masked LM(MLM)

- 기존의 conditional language model은 양방향 학습이 불가능(오직 left-to-right, right-to-left만 가능)
  -> 이는 각 단어가 간접적으로 볼 수 있게 되며, 모델이 다층 context에서 목표 단어를 쉽게 예측할 수 있음
- 심층 양방향 표현을 학습하기 위해 입력 토큰의 일부를 랜덤하게 마스킹한 후, 해당 마스킹 토큰을 예측하도록 함
- 마스킹 토큰에 해당하는 최종 은닉 벡터가 어휘 전체에 대한 softmax 출력층으로 전달됨
- 모든 실험에서, 각 시퀀스 내 WordPiece 토큰의 15%를 랜덤 마스킹함
  -> 80% 확률로 [MASK] 토큰으로 대체
  -> 10% 확률로 무작위 토큰으로 대체
  -> 10% 확률로 원래 토큰을 유지
- 그 후 T_i는 원래의 토큰을 예측하는 데 사용, cross entropy loss를 사용하여 학습

  Task #2: Next Sentence Predition(NSP)
- Question Answering과 Natural Language Inference와 같은 중요한 다운스트림 작업들은 두 문장 간의 관계를 이해하는 것에 기반
- 이러한 관계는 일반적인 언어 모델링만으로는 직접적으로 학습되지 않음
- 문장 관계를 이해하는 모델을 학습하기 위해, monoligual corpus로부터 쉽게 생성할 수 있는 이진 next sentecne prediction 과제를 사전 훈련에 포함
- 사전 훈련 샘플을 생성할 때, 문장 A와 문장 B를 선택하는 방법은
  -50%의 확률로 B는 A 다음에 오는 문장(labeled : IsNext)
  -50%의 확률로 B는 코퍼스 내에서 임의로 선택된 문장(labeled : NotNext)



- C를 다음 문장 예측에 사용한다
- 기존 연구는 문장 임베딩만을 다운스트림 과제에 전달한 반면, BERT는 모든 파라미터를 전이하여 end-task model 파라미터 초기화에 사용함

Pre-training data

- 사전 훈련 절차는 기존의 언어 모델 사전 훈련 연구를 따름
- 긴 연속된 시퀀스를 추출하기 위해 문장 단위로 셔플된 코퍼스가 아니라 문서 단위 코퍼스를 사용하는 것이 중요함

3.2 Fine-tuning BERT

- Transformer의 self-attention 메커니즘 때문에 BERT가 다양한 다운스트림 작업을 처리할 수 있음
- 적절한 입력과 출력을 교체하는 방식으로 적용
- 일반적으로 텍스트 쌍을 독립적으로 인코딩한 후 bidirectional cross attention을 적용하는 방식으로 사용
  <-> BERT는 self-attention 메커니즘을 사용하여 두 단계를 통합
  = self-attention을 사용하여 연결된 텍스트 쌍을 인코딩하면, 두 문장 간 양방향 교차 어텐션이 포함됨
- 각 작업마다 BERT 모델에 작업에 맞는 입력과 출력을 적용한 후, 모든 파리미터를 end-to-end 방식으로 미세 조정함
- Input에서 sentence A와 sentence B는 다양한 작업에 대응
  1. **패러프레이징(Paraphrasing)**: 문장 쌍(sentence pairs)
  2. **포함 관계 판별(Entailment, NLI)**: 가설(hypothesis)-전제(premise) 쌍
  3. **질문-응답(Question Answering, QA)**: 질문(question)-지문(passage) 쌍
  4. **텍스트 분류 또는 시퀀스 태깅(Text Classification / Sequence Tagging)**: 단일 텍스트-공백(∅) 쌍
- Output에서 토큰 단위 작업일 때는, token representations가 출력층으로 전달. 문장 분류 작업일 때는, [CLS] 토큰의 표현이 출력층으로 전달

6. Conclusion

- 수많은 unlabeled 데이터를 이용한 unsupervised pre-training이 NLP 분야에 큰 발전을 이룸
- deep bidirectional 구조로 확장하여, 동일한 사전 훈련 모델이 여러 NLP 작업을 효과적으로 해결